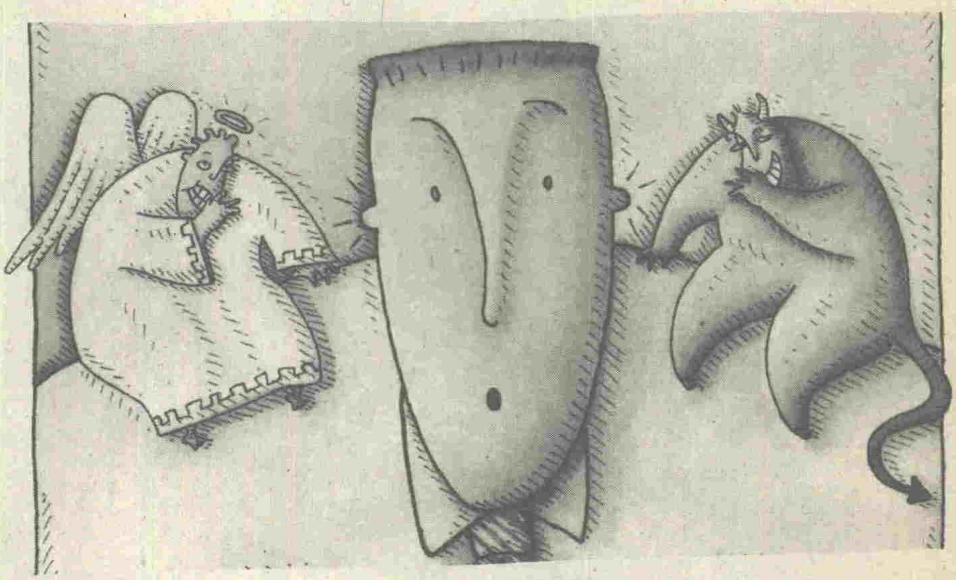
ซอฟต์แวร์
อธิการ มูลสาร

ดอสยูสเซอร์ : 19 เรื่องน่ารู้เกี่ยวกับ System

กล่าวถึงเทคนิคเล็ก ๆ น้อย ๆ ในการใช้งานคอมพิวเตอร์อย่างเต็มประสิทธิภาพ



งาน IT Trade ที่ผ่านมาผมมีโอกาสไปเยือนศูนย์ประชุมแห่งชาติสิริกิติ์อยู่ 2 วันด้วยกัน มีเรื่องที่ประทับใจผมมาก ๆ อยู่อย่างหนึ่งคือ ในส่วนของซอฟต์แวร์พาวิลเลี่ยน ตอนนี้ตลาดของวินโดวส์มาแรงมากและไม่มีทีท่าว่าจะแรงตก แพคเกจที่รันบนวินโดวส์พากันออกไทยเอดิชันตามกันมาเป็นแถว เห็นแล้วอดปลื้มไม่ได้ในความสามารถของคนไทย

หันมามองทางด้านตลาดดอสกันบ้าง จะสังเกตเห็นว่าแพคเกจจำนวนมากที่รันบนดอสหันมาใช้ระบบยูสเซอร์อินเตอร์เฟสเป็นแบบกราฟิก (GUI) เพื่อชดเชยในเรื่องของความสวยงามที่วินโดวส์แซงหน้าอยู่ จึงปรากฏแพคเกจที่เป็นกราฟิก (โดยเฉพาะโปรแกรมกรอกแบบฟอร์ม โปรแกรมโฟลว์ชาร์ต และก็ยังมีเวิร์ดโปรเซสเซอร์ สเปรดชีต ฯลฯ) ตาม ๆ กันมา

จะว่าไปแล้วมันก็เป็นประโยชน์กับยูสเซอร์เราอย่างมากที่ได้มีโอกาสเลือกใช้แต่แพคเกจดี ๆ ผมง่วนอยู่แต่ในงานนั้นจนกระทั่งดึกดื่น กลับมาถึงที่พักก็ยังรู้สึกประทับใจ เห็นแล้วมันอยากกว้านซื้อมาให้หมด

ดอสยูสเซอร์ฉบับนี้จะพูดถึงเทคนิคเล็ก ๆ น้อย ๆ ในการใช้คอมพิวเตอร์อย่างมีประสิทธิภาพ ลองสำรวจคอมพิวเตอร์ของคุณดู บางทีคุณอาจจะพบข้อบกพร่องเล็ก ๆ น้อย ๆ อันจะมีส่วนทำให้เครื่องของคุณด้อยประสิทธิภาพลงได้บ้าง จะได้แก้ไขเสีย

1 ตกแต่งไฟล์ CONFIG.SYS

เราสามารถใช้คำสั่ง REM ในไฟล์ CONFIG.SYS เพื่ออธิบายรายละเอียดต่าง ๆ ได้ ดังแสดงในรูปที่ 1 ด้วยวิธีนี้คุณสามารถเข้าใจการทำงานของแต่ละคำสั่งได้ดีขึ้น

คำสั่งนี้ใช้ได้ตั้งแต่ดอส 4 เป็นต้นไป

```
REM กำหนดขนาดของดิสค์บัฟเฟอร์เท่ากับ 3
REM เพื่อให้ทำงานร่วมกับดิสค์แคชไดรเวอร์
REM ที่ชื่อ SMARTDRV.SYS
BUFFERS=3
```

รูปที่ 1 ตัวอย่างการใช้คำสั่ง REM ในไฟล์ CONFIG.SYS

รูปที่ 1 (ต่อ)

```
REM กำหนดไฟล์แฮนเดิลเป็น 50 เผื่อ
REM ไว้สำหรับโปรแกรม DBMS
FILES = 50

REM ติดตั้งไดรเวอร์จัดการ extended
REM memory
DEVICE = C: \ DOS \ HIMEM.SYS

REM โหลดดอสไปที่ high memory area
REM และสงวน upper-memory blocks
REM ไว้สำหรับดีไวซ์ไดรเวอร์และโปรแกรม
REM เรสซิเดนต์อื่น ๆ
DOS = HIGH, UMB

REM ส่วนนี้สำหรับแบตช์ไฟล์ที่ต้องการใช้
REM escape sequence ของ ANSI.SYS
DEVICEHIGH = C: \ DOS \ ANSI.SYS

REM ติดตั้งดิสค์แคชไดรเวอร์เพิ่มประสิทธิ-
REM ภาพการทำงาน
DEVICEHIGH = C: \ DOS \ SMARTDRV.SYS
```

2 การใช้งานตัวแปร TEMP

ถ้าหากเคยใช้เครื่องหมายไปป์ (¦) เพื่อเปลี่ยนทิศทางการทำงาน เช่น DIR ¦ SORT หรือ TYPE README.TXT ¦ MORE เป็นต้น จะสังเกตเห็นการทำงานว่าค่อนข้างช้า โดยเฉพาะการทำงานกับฟลอปปี้ดิสค์ ทั้งนี้เพราะดอสจะสร้างไฟล์ชั่วคราวขึ้นมา 2 ไฟล์ด้วยกัน มันอาจจะเกิดปัญหาในกรณีที่ดิสค์นั้น Write-Protected หรือเกิดอาการ Directory Full ทำให้ดอสไม่สามารถสร้างไฟล์ชั่วคราวดังกล่าวได้

ตั้งแต่ดอส 5 เป็นต้นไป เราสามารถระบุทิศทางที่ต้องการให้ดอสไปสร้างไฟล์ชั่วคราวขึ้นที่นั่น และควรจะอยู่ที่แรมไดรฟ์ด้วยเพราะทำงานได้เร็วที่สุด ทั้งนี้ทำได้โดยการกำหนดตัวแปรเอนไวรอนเมนต์ TEMP = <ทิศทาง> เช่น

```
SET TEMP = D: \ TEMP
```

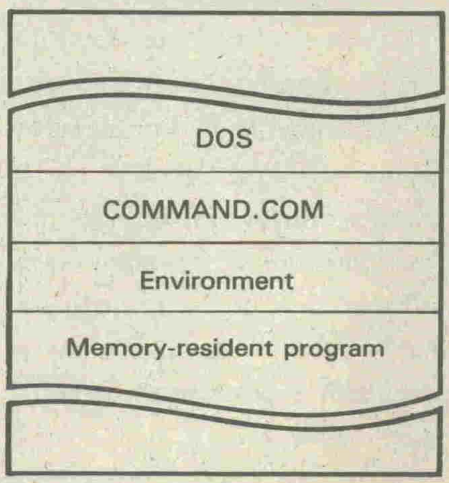
3 เพิ่มขนาดเอนไวรอนเมนต์

เอนไวรอนเมนต์ของดอสเป็นที่เก็บข่าวสารที่สำคัญ เช่น เส้นทางค้นหาคำสั่ง (PATH) ลักษณะเครื่องหมายเตรียมพร้อม (PROMPT) ทิศทางค้นหาตัวแปรคำสั่ง (COMSPEC) และอื่น ๆ

เอนไวรอนเมนต์มีขนาดเพียง 256 ไบต์ ดอสจะพยายามขยายเนื้อที่ส่วนนี้ให้ถ้าจำเป็น แต่พึงทราบไว้ว่าพื้นที่เอนไวรอนเมนต์จะต้องต่อเนื่องกันไปและอยู่ติดกับ COMMAND.COM ถ้าหากมีโปรแกรมเรสซิเดนต์ตัวใดไปสังกัดอยู่ ณ หน่วยความจำแห่งนั้น ดอสก็จะไม่สามารถขยายเอนไวรอนเมนต์ออกไปได้ และจะแสดงข้อความ "Out of environment space" (ดูรูปที่ 2)

เมื่อเกิดเหตุการณ์เช่นนี้ขึ้น คุณสามารถแก้ไขได้โดยจองขนาดเอนไวรอนเมนต์ไว้ให้มากพอ โดยใช้สวิตช์ /E ระบุขนาดเมื่อตอนที่จะโหลด COMMAND.COM ในไฟล์ CONFIG.SYS เราสามารถขยายได้ถึง 32,768 ไบต์ บรรทัดต่อไปนี้เป็นตัวอย่างการกำหนดขนาดเอนไวรอนเมนต์ให้มีขนาด 1,024 ไบต์

```
SHELL = C: \ DOS \ COMMAND.COM
C: \ DOS /P /E:1024
```



รูปที่ 2 ตำแหน่งเอนไวรอนเมนต์ในหน่วยความจำ

4

พร้อมหรือยังที่จะเรียกช่าง

ไม่ว่าจะเกิดปัญหาใด ๆ ก็ตาม ก่อนที่คุณจะเรียกช่างมาแก้ไข เพื่อป้องกันการหน้าแตก และฝึกนิสัยให้พร้อมอยู่เสมอ โปรดเตรียมข้อมูลเหล่านี้ให้พร้อม

- Serial Number ของเครื่องหรือโปรแกรมที่คุณซื้อมา
- ประเภท/เวอร์ชันและวันที่ของไบออสที่คุณใช้อยู่ ซึ่งจะดูได้จากหน้าจอเมื่อคุณเปิดสวิตช์
- ประเภทของจอภาพโมโนโครม /CGA/EGA/MCGA หรือ VGA รวมทั้งยี่ห้อของการ์ดแสดงผลและจอภาพ
- เวอร์ชันของดอสที่ใช้อยู่
- ประเภทของเม้าส์ Serial หรือ Bus และเวอร์ชันของเม้าส์ไดรเวอร์ที่ใช้อยู่
- สำเนาล่าสุดของไฟล์ CONFIG.SYS และ AUTOEXEC.BAT (ถ้าพิมพ์ออกมาทางเครื่องพิมพ์ไว้จะดีมาก)
- รายละเอียดการใช้หน่วยความจำของเครื่องที่ใช้ สำหรับดอส 4 ขึ้นไปสามารถใช้คำสั่ง MEM /CLASSIFY > PRN เพื่อพิมพ์รายละเอียดดังกล่าวได้
- จำนวนไบต์ว่างบนฮาร์ดดิสก์
- ประเภทของเครื่องพิมพ์และช่องเสียบ (Serial หรือ Parallel)
- รายละเอียดของ ERROR ที่เกิดขึ้น (รวมทั้งรายละเอียดต่าง ๆ ที่ได้กระทำก่อนหน้าที่จะเกิด ERROR ขึ้นมา)

5

อีกวิธีหนึ่งในการเพิ่มความเร็ว

เมื่อแรกเริ่มที่พีซีออกสู่ตลาดไมโครคอมพิวเตอร์ใหม่ ๆ ยุคนั้นเป็นยุคของเครื่องไมโครคอมพิวเตอร์ของแอปเปิ้ล ไม่ใช่ไอบีเอ็มอย่างทุกวันนี้ ฮาร์ดดิสก์ที่ใช้กับพีซีมีขนาดเล็กมาก ถึงขนาดว่าดอสได้บรรจุคำสั่ง VERIFY เอาไว้เป็นคำสั่งภายในเพื่อประโยชน์ในการตรวจสอบความถูกต้องในการเขียนข้อมูล โดยการนี้ I/O จะทำการอ่านข้อมูลซ้ำหลังการเขียนข้อมูลลงดิสก์ทุกครั้ง แล้วนำมาเปรียบเทียบกับข้อมูลต้นฉบับ ด้วยเหตุที่การเกิดข้อผิดพลาดมีน้อย ดอสจึงกำหนดสถานะให้ OFF แต่ถ้าต้องการใช้ความสามารถนี้ก็สามารถใช้คำสั่งต่อไปนี้ได้

```
C:\ >VERIFY ON <Enter>
```

ถึงจะดูว่าคำสั่งนี้มีประโยชน์มากในการป้องกันการเขียนข้อมูลผิดพลาด แต่มันก็มีผลข้างเคียงเช่นกัน ดิสก์ไดรฟ์นั้นเป็นกลไก เป็นเครื่องจักร มันจะทำงานช้ากว่าระบบอิเล็กทรอนิกส์ ดังนั้นถ้าคุณกำหนดสถานะ VERIFY ON ก็เท่ากับเป็นการเพิ่มหน้าที่และเพิ่มเวลาการทำงานของดิสก์ไดรฟ์แล้วลดประสิทธิภาพของเครื่องลง ผมจึงแนะนำให้ปิดสถานะนี้เสียโดยใช้คำสั่ง

```
C:\ >VERIFY OFF <Enter>
```

นอกเสียจากว่าคุณทำงานกับโปรแกรมที่ต้องวิเคราะห์ดิสก์ เช่น โปรแกรมดาต้าเบส หรือโปรแกรมแบคอัพก็ให้เปิดสถานะนี้ก่อนรันโปรแกรม แล้วจึงปิดสถานะนี้

6

ไม่สนใจ AUTOEXEC.BAT

เราสามารถระบุให้ดอสไม่เรียกแบตช์ไฟล์ AUTOEXEC.BAT มาทำงานอัตโนมัติเมื่อตอนบูตเครื่องได้ โดยเอาสวิตช์ /P ออกจากบรรทัดต่อไปนี้ในไฟล์ CONFIG.SYS

```
SHELL=C:\DOS\COMMAND.COM
C:\DOS /P
```

ข้อควรจำ

การยกเลิกการเอ็กซีคิวต์ AUTOEXEC.BAT ด้วยวิธีนี้มีข้อเสียตรงที่หากคุณใช้คำสั่ง EXIT เมื่อไรก็ตาม โดยยังอยู่ใน COMMAND.COM ก็อบปี้แรกอยู่ล่ะก็ โปรเซสเซอร์จะหยุดการทำงานทันที (Hang Up)

7

กำหนดขนาดของบัฟเฟอร์ให้ถูกต้อง

ตารางที่ 1 จะแนะนำขนาดของบัฟเฟอร์ที่ดูจะเหมาะสมกับโปรเซสเซอร์แต่ละเบอร์

ส่วนตารางที่ 2 จะแนะนำขนาด

ตารางที่ 1 แนะนำขนาดของบัฟเฟอร์ที่เหมาะสมกับโปรเซสเซอร์แต่ละเบอร์

เบอร์โปรเซสเซอร์	ขนาดของบัฟเฟอร์ที่พอเหมาะ
8088	20
80286	25
80386	30
80480	35

ตารางที่ 2 แนะนำขนาดของบัฟเฟอร์ที่เป็นค่าดีฟอลต์ของดอส

ขนาดของแรม	ค่าดีฟอลต์ที่ดอสกำหนด
128K หรือต่ำกว่า	2 (ถ้าดิสก์ขนาด 360K) 3 (ถ้าดิสก์ขนาดใหญ่กว่า 360K)
129K-255K	5
256K-511K	10
512K-640K	15

ของบัฟเฟอร์ที่เป็นค่าดีฟอลต์ของดอส โดยดูจากขนาดของแรม

แต่ถ้าใช้ดิสก์แคชไดรเวอร์ SMARTDRV.SYS ให้กำหนดขนาดบัฟเฟอร์เท่ากับ 3

8 เมื่อจำเป็นต้องใช้ FCBS

เมื่อดอสเวอร์ชันแรกปรากฏ (ปี ค.ศ. 1981) ได้ใช้วิธีการจัดการไฟล์อินพุตเอาต์พุตที่เรียกว่า File Control BlockS (บรรทัดคำสั่ง FCBS ในไฟล์ CONFIG.SYS) เป็นหลัก วิธีการดังกล่าวไม่สนับสนุนสับไดเรคทอรีตั้งแต่เวอร์ชัน 2.0 เป็นต้นมา จึงได้เปลี่ยนมาใช้วิธีการจัดการไฟล์แบบที่เรียกว่าไฟล์แฮนเดิลแทน จึงไม่น่าสงสัยเลยว่ายูสเซอร์บางท่านจะไม่รู้จักบรรทัดคำสั่งนี้

แต่เพื่อที่จะสนับสนุนโปรแกรมเก่า ๆ ดอสจึงอนุญาตให้มีคำสั่งตั้งค่าระบบ FCBS ในไฟล์ CONFIG.SYS ดังเดิมได้ด้วย ดังนั้นถ้าไม่มีโปรแกรมสำเร็จรูปเก่า ๆ (ก่อนปี ค.ศ. 1985) ใช้งานแล้วก็ให้ลบคำสั่ง FCBS ออกจากไฟล์ CONFIG.SYS เสีย แต่หากคุณมีโปรแกรมเก่าเป็นจำนวนมากที่ไม่สามารถเปิดไฟล์ใช้ในแบบไฟล์แฮนเดิลได้ ให้เติมบรรทัดต่อไปนี้ในไฟล์ CONFIG.SYS

```
FCBS=16
```

9 ถ้าเกิด CRITICAL ERROR

เมื่อเกิดข้อผิดพลาดอันจะเนื่องมาจากการเขียนและอ่านข้อมูล หรือข้อผิดพลาดทางฮาร์ดแวร์ใด ๆ คุณคงเคยคุ้นกับข้อความต่อไปนี้

```
Abort, Retry, Fail?
```

แต่ถ้าคุณบรรจุสวิตช์ /F เพิ่มเติมเข้าไปในไฟล์ CONFIG.SYS ดังตัวอย่างต่อไปนี้ ดอสจะถือเสมือนหนึ่งคุณได้เลือกออปชัน Fail อัตโนมัติ

```
SHELL=C:\DOS\COMMAND.COM
C:\DOS /P /F
```

10 บวกลบเลขฐานสิบหกด้วยดีบัก

บ่อยครั้งที่เราต้องพบเจอกับเลขฐานสิบหก โปรแกรมดีบักเป็นโปรแกรมที่มีประโยชน์มากคือ นอกจากจะใช้แก้บักอันสิงสถิตย์อยู่ในโปรแกรมของเราด้วย ก็ยังสามารถใช้สร้างไฟล์คำสั่งด้วยภาษาแอสเซมบลีอย่างง่ายได้อีกด้วย ประโยชน์ของดีบักอีกประการหนึ่งที่เราจะได้ทราบกันตรงนี้ก็คือว่า ดีบักบวกลบเลขฐานสิบหกได้

ลองป้อนคำสั่งตัวอย่างต่อไปนี้ที่เครื่องหมายเตรียมพร้อมของดีบัก แล้วลองคำนวณด้วยกระดาษอีกทีว่าได้ผลลัพธ์ตรงกันไหม

```
-HOFEOF
010D 00EF
```

ไฮเฟน (–) เป็นเครื่องหมายเตรียมพร้อมของดีบัก คำสั่ง H (Hexadecimal) จะทำการบวกและลบเลขฐานสิบหกสองตัวอันได้แก่ FE และ OF ผลลัพธ์ที่ได้นั้นตัวแรกคือ ผลบวก ตัวที่สองคือ ผลต่าง นี่คือเหตุผลว่าทำไมผมจึงบอกว่าดีบักบวกลบเลขฐานสิบหกได้ ก็เพราะว่ามันบวกลบได้จริง ๆ น่ะซิ

11

คำสั่ง FOR ใช้ที่ดอสพรอมต์ก็ได้

ไม่ใช่ว่าจะใช้แต่ในแบตช์ไฟล์เท่านั้น คำสั่ง FOR ที่เป็นคำสั่งแบตช์-ไฟล์เราสามารถใช้ที่ดอสพรอมต์ก็ได้ จากตัวอย่างต่อไปนี้เราอยากจะดูเท็กซ์-ไฟล์หลาย ๆ ไฟล์พร้อมกัน และอยากจะดูเฉพาะไฟล์บางไฟล์ด้วย ได้แก่ CONFIG.SYS และ AUTOEXEC.BAT จะใช้คำสั่ง TYPE *.TXT และ TYPE CONFIG.SYS AUTOEXEC.BAT ฯลฯ ก็ไม่ได้ ลองพิมพ์คำสั่งต่อไปนี้แล้วดูว่าผลที่ได้ตรงตามที่เราต้องการหรือไม่

```
C:\>FOR %A IN (*.TXT) DO TYPE %A
```

และ

```
C:\>FOR %B IN (CONFIG.SYS AUTOEXEC.BAT) DO TYPE %b
```

สะดวกขึ้นเยอะจริงไหมครับ ดอสจะเอาค่าในเซตที่เรากำหนดในเครื่องหมายวงเล็บขยายออกมาทีละตัว แล้วนำไปใส่ตัวแปร A และ B ตามลำดับ (ตัวแปรได้แก่ A-Z, และใช้ซ้ำได้) คำสั่ง TYPE ที่ตามหลังมาก็เพียงแต่นำเอาค่าในตัวแปรดังกล่าวไปใช้เท่านั้นเอง แต่ถ้าใช้ในแบตช์ไฟล์อย่าลืมว่าต้องใช้เครื่องหมายเปอร์เซ็นต์ (%) ควบกัน 2 ตัวนะครับ

อีกตัวอย่างหนึ่งสมมติว่ามีไฟล์ที่ชื่อ A1-A10.TXT รวมแล้ว 10 ไฟล์ แต่ผมต้องการจะเลือกก๊อบปี้เฉพาะไฟล์ A1.TXT, A5.TXT, A6.TXT และ A10.TXT ผมก็อาศัยเซตนี่แหละช่วยโดยพิมพ์คำสั่งดังนี้

```
C:\>FOR %F IN (1, 5, 6, 10) DO COPY
    C:\MYTEXT\A%F.TXT  A:
```

12

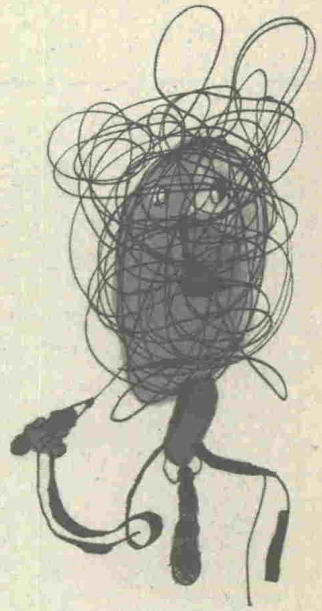
ANSI.SYS ทำให้ I/O ช้าลง

ถึงแม้ว่า ANSI.SYS จะมีประโยชน์ในการทำงานมาก เช่น กำหนดหน้าที่การทำงานของคีย์บอร์ด กำหนดสีจอภาพ เปลี่ยนโหมดการแสดงผล เลื่อนตำแหน่งเคอร์เซอร์ ฯลฯ แต่มันจะทำการตรวจสอบคีย์บอร์ดทุกครั้งที่กด และตรวจสอบตัวอักษรทุกตัวที่แสดงผลผ่านดอส ถ้าไม่จำเป็นต้องใช้ความสามารถของ ANSI.SYS ก็ไม่ต้องติดตั้งลงไป

13

COMMAND.COM ฉบับใหม่ รวดเร็วทันใจ

COMMAND.COM เป็นตัวแปรคำสั่งของดอส ซึ่งจะบรรจุคำสั่งภายในของดอสไว้ และทำหน้าที่จัดการกับคำสั่งที่คุณป้อนลงที่ดอสพรอมต์ เพื่อเป็นการประหยัดเนื้อที่ในหน่วยความจำ ดอสจึงแบ่ง COMMAND.COM ออกเป็น 2 ส่วนคือ ส่วนที่อยู่ถาวร (Resident Part) และส่วนชั่วคราว (Transient Part) (ดังแสดงในรูปที่ 3)

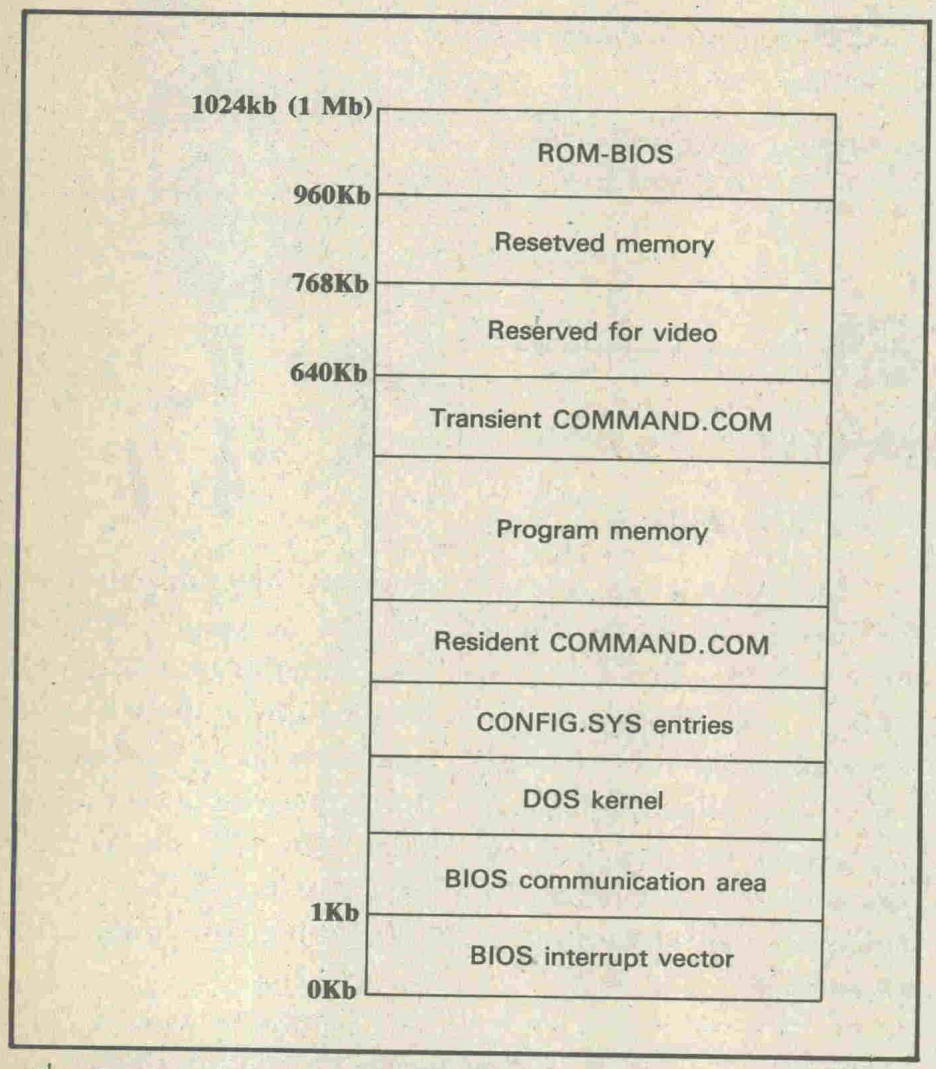


ส่วน Transient หรือส่วนที่อยู่ชั่วคราวอาจถูกโปรแกรมซึ่งมีขนาดใหญ่โหลดทับลงมาก็ได้ เมื่อเกิดเหตุการณ์เช่นนี้ขึ้น ส่วนที่อยู่ถาวรจะต้องทำการโหลด COMMAND.COM ลงมาใหม่ โดยอาศัยทิศทางที่กำหนดไว้ในตัวแปรระบบ COMSPEC ซึ่งได้มาจากบรรทัดคำสั่ง "SHELL=" ในไฟล์ CONFIG.SYS อีกที

ถ้าคุณมีหน่วยความจำเหลือ ไม่ว่าจะเป็น EXTENDED หรือ EXPANDED ก็ตาม แบ่งหน่วยความจำดังกล่าวมาสร้างเป็นแรมไดรฟ์ ก๊อบปี้ไฟล์ COMMAND.COM ไปไว้ที่นั่น กำหนดให้ตัวแปรระบบ COMSPEC เก็บทิศทางไปที่ไดเรคทอรีบนแรมไดรฟ์ เพียงเท่านี้ดอสก็สามารถโหลด COMMAND.COM มาใหม่ได้และเร็วกว่าเดิมมาก

เพิ่ม 2 บรรทัดต่อไปนี้เข้าไปในไฟล์ AUTOEXEC.BAT ผมขอสมมติให้แรมไดรฟ์ที่คุณมีอยู่เป็นไดรฟ์ D

```
COPY C:\DOS\COMMAND.COM D:\
SET COMSPEC=D:\COMMAND.COM
```



รูปที่ 3 ตำแหน่งของส่วนต่างๆ ของ COMMAND.COM ในหน่วยความจำ

14

รู้จักกับคำสั่ง LASTDRIVE

ดอสรู้จักไดรฟ์อยู่สองแบบ แบบแรกเรียกว่า ฟิสิคัลไดรฟ์ (Physical Drive) ซึ่งก็คือดิสก์ไดรฟ์ที่เราเห็น ๆ กัน เช่น ไดรฟ์ A และ C ส่วนแบบที่สองเรียกว่า โลจิคัลไดรฟ์ (Logical Drive) คุณจับต้องฟิสิคัลไดรฟ์ได้ แต่ไม่สามารถจับต้องโลจิคัลไดรฟ์ได้ เพราะมันคือตัวอักษรที่ใช้อ้างอิงถึงดิสก์ไดรฟ์ ไดเรคทอรี แล้วก็แรมไดรฟ์ คำสั่ง SUBST ที่แทนชื่อไดเรคทอรีด้วยไดรฟ์ A-Z เป็นอีกตัวอย่างของโลจิคัลไดรฟ์

ค่าดีฟอลต์ที่ดอสกำหนดให้โลจิคัลไดรฟ์คือ A ถึง E แต่เราสามารถกำหนดเองได้ โดยระบุไปในบรรทัด "LASTDRIVE=" ในไฟล์ CONFIG.SYS คุณสามารถกำหนดได้ถึงตัวอักษร Z การกำหนด LASTDRIVE ไม่มีผลเสียใดนอกจากจะกินหน่วยความจำไป 80-96 ไบต์ต่อหนึ่งโลจิคัลไดรฟ์เท่านั้น ถ้าคุณจะกำหนดถึงไดรฟ์ Z เลยก็ไม่มีใครว่า แต่ถ้าไม่ได้ใช้ผมว่าน่าจะกำหนดแค่พอดีจะดีกว่านะ

15

เพิ่มขนาดเอนไวรอนเมนต์ (อีกแล้ว)

การเพิ่มขนาดเอนไวรอนเมนต์เท่าที่เราพอจะทราบกันอยู่ ทำได้โดยการเพิ่มสวิตช์ /E:size เข้าไปในบรรทัด "SHELL=" ในไฟล์ CONFIG.SYS (ดูจากการเพิ่มขนาดเอนไวรอนเมนต์ข้างต้น) อย่างไรก็ตาม เรายังมีวิธีอีกวิธีหนึ่ง แต่ก่อนอื่นผมจะขอพูดถึงข้อเสียของวิธีแรก การเพิ่มขนาดเอนไวรอนเมนต์โดยวิธีแรกนั้น ดอสจะจองพื้นที่ของหน่วยความจำเป็นจำนวนไบต์เท่ากับขนาดที่กำหนด ในการจองครั้งนี้ดอสจะถือเป็นการจองถาวรคือ ถ้าไม่ปิดเครื่องก็ไม่มีการคืนหน่วยความจำส่วนนี้ให้เด็ดขาด ส่วนใหญ่แล้วที่เราต้องการขยายขนาดเอนไวรอนเมนต์ก็เพื่อให้แบตช์ไฟล์ของเราใช้ตัวแปรเอนไวรอนเมนต์ (ผมมักใช้คำว่าตัวแปรระบบแทน) ได้เต็มที่ บางทีก็จองเผื่อไว้เยอะมาก เช่น /E:1024, /E:2048 ฯลฯ

ผมเป็นคนหนึ่งที่จองเอาไว้ 1,024 ไบต์หรือ 1 กิโลไบต์ ซึ่งอาจจะไม่เพียงพอด้วยซ้ำ ในกรณีที่แบตช์ไฟล์ของผมใช้ตัวแปรระบบมาก พักหลัง ๆ นี้ผมจึงจองเอาไว้ถึง 2,048 ไบต์หรือ 2 กิโลไบต์

ก็ไม่น่าจะมีปัญหาอะไร (หน่วยความจำเพียงแค่นี้นับว่าน้อยมากเพราะเครื่องไมโครฯ เกือบทั้งหมดมีหน่วยความจำ 640 กิโลไบต์ขึ้นไป) แต่ผมมาพบภายหลังว่าแบตช์ไฟล์ส่วนใหญ่แล้วจะใช้ตัวแปรระบบไม่เกิน 512 ไบต์หรือ 1/2 กิโลไบต์ ซึ่งก็คือค่าดีฟอลต์ที่ดอสกำหนดไว้พอเหมาะพอเจาะอยู่แล้ว จึงนึกเสียดายหน่วยความจำที่ไม่ได้ใช้นั้นขึ้นมาจับใจ ก็มานึกถึงพารามิเตอร์หรือสวิตช์ของไฟล์ COMMAND.COM อีกตัวหนึ่งขึ้นมาได้ นั่นก็คือ /C

สมมติว่าผมมีแบตช์ไฟล์ที่ใช้ตัวแปรระบบมากหน่อย ต้องกันเอนไวรอนเมนต์ไว้สัก 2,048 กิโลไบต์จึงจะพอ จะไปเซตในไฟล์ CONFIG.SYS ก็เห็นจะไม่สะดวกนักเพราะถ้าผมต้องการหน่วยความจำส่วนนี้คืน ผมก็ต้องไปแก้ในไฟล์ CONFIG.SYS แล้วบูตดอสกันอีกรอบ ชะรอยผมจะต้องยอมเปลืองหน่วยความจำเพื่อโหลดเอา COPY ที่สองของ COMMAND.COM ลงไปแล้วเอ็กซีคิวต์แบตช์ไฟล์ดังกล่าวผ่าน COMMAND.COM COPY ที่สองนี่แหละเมื่อผมโหลด COMMAND.COM ได้ ผมก็เซตขนาดเอนไวรอนเมนต์ให้กับ COMMAND.COM COPY ที่สองนี้ด้วยพารามิเตอร์ /E:size ได้เช่นกัน (ท้าวความถึง "เพิ่มขนาดเอนไวรอนเมนต์" ข้างต้น ตำแหน่งของพื้นที่เอนไวรอนเมนต์จะต้องอยู่ติดกับ COMMAND.COM เสมอ ซึ่งรับกันพอดีกับวิธีนี้) นี่ก็คืออีกวิธีหนึ่งที่ผมพูดถึงซึ่งทั้งหมดพอจะเขียนออกมาเป็นขั้นเป็นตอนดังนี้

1: ป้อนคำสั่ง "COMMAND /E:2048" เพื่อเรียก COMMAND.COM COPY ที่สอง พร้อมกับกำหนดขนาดเอนไวรอนเมนต์เป็น 2,048 ไบต์

2: เรียกใช้แบตช์ไฟล์ดังกล่าว

3: ป้อนคำสั่ง "EXIT" เพื่อออกจาก COMMAND.COM COPY ที่สองและคืนหน่วยความจำรวมทั้งพื้นที่เอนไวรอนเมนต์ให้กับดอส

ค่อนข้างจะยุ่งยากสักเล็กน้อย สวิตช์ /C ที่ผมพูดถึงจะช่วยให้วิธีนี้ง่ายขึ้นมาก หน้าที่ของ /C นี้ก็คือกำหนดให้ COMMAND.COM เอ็กซีคิวต์ไฟล์คำสั่ง (.COM, .EXE, .BAT) แล้วคืนหน่วยความจำให้กับดอสเป็นขั้นตอนเดียว เราจึงอาจเขียนขั้นตอนทั้ง 3 ข้างต้นด้วยคำสั่งเพียงสองบรรทัดดังนี้

```
C:\>COMMAND /E:2048 /C
        ชื่อแบตช์ไฟล์หรือไฟล์ที่จะเอ็กซีคิวต์
```

ดอสเวอร์ชัน 2.0 ถึงเวอร์ชัน 3.2 ไม่มีคำสั่ง CALL ในแบตช์ไฟล์ เมื่อเรียกชื่อแบตช์ไฟล์จึงเป็นการกระโดดไปเอ็กซีคิวต์แบตช์ไฟล์นั้น ๆ โดยไม่กลับมาอีกเลย จากที่ได้กล่าวมาข้างต้น ถ้าคุณใช้ดอสเวอร์ชันระหว่าง 2.0-3.2 คุณก็สามารถใช้พารามิเตอร์ /C นี้แทนคำสั่ง CALL ได้ เสร็จจากแบตช์ไฟล์ที่ถูกเรียกก็จะกลับมาทำงานบรรทัดคำสั่งถัดไป

16

เพิ่มความเร็วต้องลดการตรวจ Ctrl-Break

ถ้าเรากำหนดสถานะ BREAK ให้ ON แล้วดอสจะต้องทำงานมากขึ้นเพื่อตรวจการกดปุ่ม Ctrl-C หรือ Ctrl-Break อยู่ตลอดเวลา ซึ่งมีผลให้เวลาในการประมวลผลเพิ่มมากขึ้น ถ้าไม่จำเป็นต้องใช้ความสามารถนี้ให้เซต OFF

17

ไดรเวอร์ล่าสุดมาแล้ว

HIMEM.SYS และ EMM386.EXE เป็นไดรเวอร์จัดการหน่วยความจำที่ใช้ควบคู่กัน มีเรื่องสับสนอยู่เล็กน้อยเนื่องจากทั้งสองไฟล์ดังกล่าวมีทั้งเวอร์ชันที่มากับดอส 5.0 และ 6.0, Windows 3.0 และ 3.1 เรียกว่าถ้าขึ้นชื่อแพคเกจของไมโครซอฟต์ล่ะก็ต้องแถมไดรเวอร์ทั้งสองมาแล้วก็เป็นเวอร์ชันล่าสุดเสมอ ใช้คำสั่ง DIR \HIMEM.SYS /S และ DIR \EMM386.EXE /S เพื่อค้นหาไฟล์ที่ชื่อเดียวกันแต่มีหลายเวอร์ชันอยู่ต่างไดเรคทอรีกันไป ให้สังเกตวันที่อัพเดตครั้งล่าสุดของไฟล์ก็จะทราบได้ว่าไดรเวอร์ไหนใหม่สุด จากนั้นก็แก้ไข CONFIG.SYS ให้เรียกใช้ไดรเวอร์เวอร์ชันล่าสุด จะดีที่สุดและมีประสิทธิภาพแบบสุด ๆ กันไปเลย

18

Undocumented Entry in CONFIG.SYS

เราได้รู้ถึงวิธีการใช้คำสั่ง REM ในไฟล์ CONFIG.SYS เพื่ออธิบายวัตถุประสงค์ของแต่ละบรรทัด (ดู "ตกแต่งไฟล์ CONFIG.SYS") ยังมีอีกหนทางหนึ่งที่จะใส่คอมเมนต์ในไฟล์ CONFIG.SYS นั่นก็คือใช้คำสั่ง "COMMENT=" (คำสั่งนี้ไม่ปรากฏอยู่ในคู่มือใด ๆ หรือที่เรียกว่า Undocumented) ซึ่งสนับสนุนเฉพาะดอสเวอร์ชัน 2.0 ถึงเวอร์ชัน 5.0 เท่านั้น

"COMMENT=" เป็นคำสั่งที่ระบุให้ดอสถือว่ากลุ่มของอักขระใด ๆ ก็ตามที่จะกำหนดต่อไปนี้ ไม่ว่าจะพบอยู่ที่ต้นบรรทัดใดในไฟล์ CONFIG.SYS ให้ถือว่าบรรทัดนั้นทั้งบรรทัดเป็นคอมเมนต์ อ่านแล้วงง ๆ อยู่บ้างใช่ไหมครับ ตัวอย่างในรูปที่ 4 จะทำให้เข้าใจได้กระจ่างขึ้น โดยจะกำหนดให้คำว่า NOTES แทนคอมเมนต์

ทำไมจึงต้องกำหนดเช่นนี้ในเมื่อมีคำสั่ง REM อยู่แล้ว (ในที่นี้คำว่า REM หรือ Remark กับคำว่า COMMENT มีความหมายเหมือนกัน) ในกรณีที่คุณไม่ต้องการติดตั้งดีไวซ์ไดรเวอร์คือ ยกเลิกไดรเวอร์ทุกตัว คงจะไม่สะดวกนักหากคุณจะใส่คำสั่ง REM นำหน้าทุกบรรทัดที่ขึ้นต้นด้วยคำว่า DEVICE ด้วยคำสั่ง COMMENT นี้คุณสามารถกำหนดง่าย ๆ ว่า COMMENT=DEVICE เพียงเท่านี้ดอสก็จะมองคำว่า DEVICE เป็นคอมเมนต์และข้ามการทำงานไป และยังสามารถย่อให้เหลือสั้น ๆ ลงเหลือเพียง COMMENT=DE ก็ยังถือว่าใช้ได้เพราะใน CONFIG.SYS คำสั่งที่ขึ้นต้นด้วย "DE" มีเพียงคำสั่ง DEVICE และ DEVICEHIGH เท่านั้น

```
COMMENT=NOTES
NOTES Specify the number of file handles
FILES=30
NOTES Specify the number of disk buffers
BUFFERS=25
```

รูปที่ 4 การใช้คำสั่ง COMMENT

```
@ECHO OFF
SET SAVE_PROMT=%PROMPT%
PROMPT Type EXIT to return to DOSSHELL$_$P$G
DOSSHELL
PROMPT %SAVE_PROMPT%
SET SAVE_PROMPT=
```

รูปที่ 5 DSHELL.BAT เปลี่ยนเครื่องหมายพรอมต์เป็นข้อความเหมือน

19

เตือนกันหน่อย...น่ะ

บ่อยครั้งที่คุณ Shell ออกมาจากโปรแกรมใด ๆ แล้วลืม ถ้าจะเปลี่ยนพรอมต์สักหน่อยก็คงจะเตือนความจำเราได้ดีทีเดียว แบตช์ไฟล์ต่อไปนี้ (รูปที่ 5) จะเรียกใช้ DOSSHELL ในกรณีที่คุณ Shell ออกมาจาก DOSSHELL เครื่องหมายพรอมต์จะเปลี่ยนไปเป็นข้อความเตือน จนกว่าจะออกจาก DOSSHELL เป็นที่เรียบร้อยแล้วจึงจะคืนเครื่องหมายพรอมต์ให้เหมือนเดิม

เมื่อเรียกใช้แบตช์ไฟล์ เครื่องหมายพรอมต์จะเปลี่ยนเป็น

```
Type EXIT to return to DOSSHELL
C:\>_
```